Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Vasil Krstev

Napovedovanje uspešnosti nogometnih ekip z uporabo analize omrežij

diplomsko delo

univerzitetni študijski program prve stopnje
računalništvo in informatika

Mentor: doc. dr. Lovro Šubelj

Ljubljana, 2016

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Sodobna analiza omrežij nudi številne pristope za proučevanje zgradbe velikih kompleksnih omrežij. V diplomskem delu analizirajte omrežje nogometnih ekip angleške lige. Predstavite metode za določanje pomembnosti vozlišč takega omrežja ter le-te uporabite za napovedovanje uspešnosti ekip v izbranem časovnem obdobju. Primerjajte uspešnost izbranih metod, kritično ovrednotite rezultate ter podajte predloge za nadaljnje delo.

Izjava o avtorstvu diplomskega dela

Spodaj podpisani Vasil Krstev, z vpisno številko **63120371** sem avtor diplomskega dela z naslovom:

Napovedovanje uspešnosti nogometnih ekip z uporabo analize omrežij

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Lovra Šublja ,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 5. septembra 2016 Podpis avtorja:

Za vso strokovno podporo, ideje in pomoč pri izdelavi diplomskega dela bi se zahvalil doc. dr. Lovru Šublju. Za vso moralno in finančno podporo, ki sta mi jo nudila tekom študij, se moram zahvaliti materi Tatjani in očetu Zoranču. Hkrati se zahvaljujem vsem kolegom in prijateljem za motivacijo ter podporo tekom celotnega šolanja.

"At the end of this game, the Champions League Trophy will be only six feet away from you, and you'll not even able to touch it if we lose. And for many of you, that will be the closest you will ever get. Don't you dare come back in here without giving your all." - Sir Alex Ferguson

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
GPS	Global Positioning System	Globalni sistem pozicioniranja
BC	Betweenness Centrality	Središčnost vmesnosti
CC	Closeness Centrality	Bližinska središčnost
EC	Eigenvector Centrality	Središčnost lastnega vektorja
PR	PageRank	Uvrstitev strani
AUTH	Authorities	Viri
HUB	Hubs	Kazala
MSE	Mean Squared Error	Povprečna kvadratna napaka
MAE	Mean Absolute Error	Povprečna absolutna napaka
RSE	Relative Squared Error	Relativna kvadratna napaka
RAE	Relative Absolute Error	Relativna absolutna napaka
CRISP-DM	Cross Industry Standard Process for Data Mining	Industrijski procesni model za podatkovno rudarjenje
k-NN	k-Nearest Neighbors	k-najbližjih sosedov
HITS	Hyperlink-Induced Topic Search	HITS

LR	Linear Regression	Linearna regresija
SVM	Support Vector Machine	Metoda podpornih vektorjev
RT	Regression Tree	Regresijsko drevo
RF	Random Forest	Naključni gozd
UEFA	Union of European Football Associations	Združenje evropskih nogometnih zvez

Povzetek

Cilj diplomske naloge je podrobna analiza nogometnih tekem s pomočjo analize omrežij ter gradnja napovednega modela za napovedovanje lastnosti tekem na podlagi atributov dobljenih iz tovrstnega omrežja. Področji analize omrežij in podatkovnega rudarjenja postajata vse bolj priljubljeni področji v računalniškem svetu za odkrivanje znanj iz podatkov. V sodobnem svetu se opravljajo meritve tudi pri nogometnih tekmah, zato smo se odločili, da to področje podrobneje raziščemo in analiziramo s pomočjo analize omrežij ter poskušamo čim natančneje napovedati lastnosti same tekme kot so število točk, golov, kartonov, kotov in drugo.
V nalogi so najprej podrobneje predstavljene metode in tehnike analize omrežij. Nato so predstavljeni vsi algoritmi in mere uspešnosti, ki jih uporabljamo pri napovedi. Sledi predstavitev podatkov, krajši primer delovanja odkrivanja skupnosti ter potek gradnje napovednega modela. Sledi interpretacija dejanskih napovedi ter primerjava učinkovitosti napovedne točnosti. Pri testiranju smo preverili napovedi za vse lastnosti dobljene iz analize omrežij, predstavili pa smo le tiste, ki so najbolj natančno napovedale ciljno spremenljivko. Za zaključek smo na kratko primerjali rezultate ter izpostavili glavne pomanjkljivosti. Podali smo tudi navodila za nadaljnje delo ter smernice kako napovedni model lahko še izboljšamo.

Ključne besede: nogomet, analiza omrežij, podatkovno rudarjenje, napovedovanje.

Abstract

The goal of this thesis was a detailed analysis of football matches through network analysis and building a predictive model for predicting characteristics of matches based on attributes obtained from such a network. The fields of network analysis and data mining are becoming increasingly popular in the computing world for data knowledge discovery. In the modern world, people are making a lot of measurements on football matches, so we decided to investigate this area in detail, analyse it through network analysis and try to most accurately predict the characteristics of a single game such as the number of points, goals, cards, corners and other.

The thesis first presents the methods and techniques for network analysis. Then the algorithms and measures of performance are presented, which are used for the prediction. Next, we present the data, give an example of application of community detection and present the process of building predictive models. What follows is the actual interpretation of predictions and comparison of the effectiveness of predictive accuracy. We have examined the prediction strength for all of the properties obtained from network analysis, but we present only those that gave best results. In conclusion, we briefly compare the results and highlight the main weaknesses. We present possible directions for future work and give guidance on how the predictive model can be further improved.

Keywords: football, network analysis, data mining, prediction.

Poglavje 1

Uvod

Kaj razumemo pod besedo šport? Šport kot veja se je začel razvijati 4000 let nazaj in je dejavnost, ki zahteva fizičen napor in veščine, v katerih posameznik ali skupina tekmuje drug proti drugemu. Danes je med pomembnejšimi športi zagotovo nogomet, ki pa ni zgolj šport, ampak s časoma postaja dejavnost, ki povezuje ljudi, ne glede na spol, religijo in barvo kože.
Nogomet kot šport se je prvič pojavil v Angliji v sredini devetnajstega stoletja. S časoma se je razvijal in postajal vse bolj priljubljen med ljudmi. V času prve in druge svetovne vojne so bile skoraj v vseh državah vse športne lige odpovedane razen nogometnih, kar pomeni, da je bil nogomet že takrat zelo priljubljen. Nogomet je skupinski šport, v katerem tekmujeta dve ekipi, sestavljenih iz 11 igralcev. Tekma traja 90 minut in ekipa, ki doseže več golov je zmagala, če pa se zgodi, da sta ekipi dosegli enako število golov, potem je tekma neodločena. Danes obstajajo različna tekmovanja, kot so ligaška, evropska in svetovna na ravni ekipe in države ter imajo različene sisteme tekmovanja.
V sodobnem nogometu vse ekipe veliko časa posvetijo analizi tekem. Pridobivanje podatkov poteka preko GPS čipa, ki je vgrajen v kopačkah igralcev (nogometni čevlji) in posebnih programskih opremah, ki pa niso prosto dostopne. S pomočjo njih beležijo raznovrstne podatke, katere se potem uporabi za statistične analize, tako za posameznike, kot tudi za celotno ekipo. Da-

nes profesionalne ekipe uporabljajo računalniške tehnike, zato da čim hitreje pridobijo veliko podatkov o naslednjem nasprotniku. Z dobljenimi podatki poglobljeno analizirajo nasprotnika in si zagotovijo morebitno dodatno prednost. Težko je dobiti takšne podatke oziroma bi jih morali plačati, zato smo se odločili, da bomo analizirali zgolj ekipe. Za nas je bil to velik izziv, saj do sedaj obstaja zelo malo tovrstnih analiz.

Ena izmed tehnik, s katerimi se analizirajo tekme je t.i. področje analize omrežij. Kot veja računalništva se hitro širi in počasi postaja ena od pomembnejših področij za odkrivanje znanj iz podatkov. Veja je podobna kot teorija grafov, saj se podatki preoblikujejo v vozlišča ter v povezave med njimi. Omrežja so lahko velika z več tisoč povezav in vozlišč, lahko pa so majhna le z nekaj vozlišči in povezav med njimi. Naše omrežje spada med majhna omrežja, kjer vozlišča pomenijo ekipe, povezava pa nam pove, da je ena ekipa premagala drugo ekipo.

V današnjem nogometnem svetu imajo značilne vloge tudi športne stavnice, ki predlagajo kvote, na katere ljudje stavijo denar, s čimer tudi dodatno zaslužijo. Obstaja veliko ljudi, ki se s tem ukvarjajo tudi profesionalno, izkoriščajo svoje ekspertno znanje in poiskušajo čim natančneje napovedati izid tekme. Za ta namen poleg analize tekme na podlagi analize omrežij, cilj naloge je bil zgraditi napovedni model s pomočjo lastnostih iz analize omrežij. V diplomski nalogi bomo poskušali zgraditi model, ki bo napovedoval: število točk, število golov, število prejetih kartonov, število kotov ekipe ipd. Pri napovedovanju si bomo pomagali z računalniškimi tehnikami iz področja kot je podatkovno rudarjenje, zato ker je močno povezano z analizo omrežij, saj obe temeljita na odkrivanju znanj iz podatkov.

Napovedi bodo izvedene na podlagi podatkov iz angleške lige, saj je ena od najbolj gledanih, obiskovanih in zanimivih lig na svetu [7]. V 2. poglavju smo podrobno opisali metode in tehnike iz analize omrežij in podatkovnega rudarjenja, s katerimi si bomo pomagali za analizo in napovedi. V 3. poglavju smo na kratko še opisali orodja, s katerimi smo se ukvarjali tekom diplomske naloge. Nato smo v 4. poglavju podali podatke, rezultate ter njihovo

interpretacijo. Na koncu v 5. poglavju pa smo opisali sklepne ugotovitve in možnosti za nadaljne delo.

Poglavje 2

Metode in tehnike

V nadaljevanju sledi pregled področja analize omrežij ter glavne značilnosti podatkovnega rudarjenja.

2.1 Analiza omrežij

Začetki analize omrežij segajo v osemnajsto stoletje z rešitvijo problema sedmih mostov Königsberg od strani Leohnard Euler. Zaradi zanimivosti je področje sčasoma postajalo vse bolj priljubljeno za reševanje tovrstnih problemov. Danes obstajajo različne vrste omrežij kot so [3]:

- socialna omrežja, ki preučujejo družbene relacije med ljudmi ali skupinami ljudmi kot so na primer prijateljstva. Te vrste omrežij so najbolj raziskovana, najbolj znana med njimi sta v sodobnem svetu Facebook in Twitter ;

- biološka omrežja, ki se pogosto uporabljajo v mnogih vejah biologije, kot primerna predstavitev vzorcev za interakcije med ustreznimi biološkimi elementi. Na primer, molekularni biologi uporabljajo omrežja, s čimer bi predstavili vzorce kemičnih reakcij med kemikalijami v celicah, medtem ko jih nevro znanstveniki uporabljajo za predstavitev vzorčnih povezav med možganskimi celicami ;

- informacijska omrežja, ki so sestavljena iz različnih medsebojno povezanih podatkov. Najbolj znan primer je svetovni splet, čeprav obstaja tudi mnogo drugih primerov, ki so primerni za raziskovanje. Ta pristop lahko uporabimo pri modeliranju zvez citiranj v znanstvenih člankih, kjer vozlišča predstavljajo posamezni članki, povezave so pa usmerjene v smeri citiranih vozlišč .

V nadaljevanju sledi splošen pregled analize omrežij s poudarkom na značilnostih, katere smo uporabljali za doseganje naših ciljev.

2.1.1 Splošne značilnosti

Omrežje [1] si lahko predstavljamo kot urejen par $G=(V,E)$, ki vsebuje množico vozlišč V(2.1) ter množico povezav E(2.2) med njimi.

$$V = \{v_1, v_2, ..., v_n\} \tag{2.1}$$

$$E \subseteq \{\{v_i, v_j\} | v_i, v_j \in V\} \tag{2.2}$$

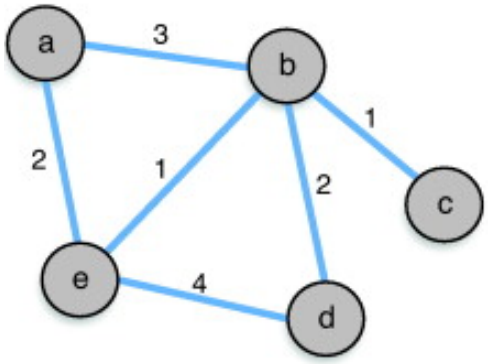
Matematično gledano, omrežje lahko predstavimo z matriko sosednosti A. To je $n \times$ n kvadratna matrika, kjer je n število vozlišč v omrežju. Elementi matrike sosednosti so predstavljeni kot:

$$A_{ij} = \begin{cases} 1; & \text{če obstaja povezava med } i \text{ in } j \\ 0; & \text{če ne obstaja povezava med } i \text{ in } j \end{cases} \tag{2.3}$$

Vrednost A_{ij} je lahko tudi večja glede na število povezav, ki obstajajo med vozliščema i in j oziroma njihova utež.
V nekaterih omrežjih imajo lahko ene povezave večji pomen kot druge. Takim omrežjim pravimo utežena omrežja, kjer ima vsaka povezava določeno

[1] Zaradi boljša preglednost v nadaljevanju uporabimo izraz omrežje, tudi v primerih, kjer je bolj primerno uporabljati izraz graf.

pomembnost oziroma utež. Omrežja so lahko usmerjena (angl. directed network), kjer ima vsaka povezava med vozlišči smer, ki je prikazana s puščico, glede na smer ali neusmerjena (angl. undirected network), kjer povezava nima smeri (je navadna črtica) [16]. Na sliki 2.1 in sliki 2.2 sta prikazani dve enostavni omrežji z njihovimi matriki sosednosti.



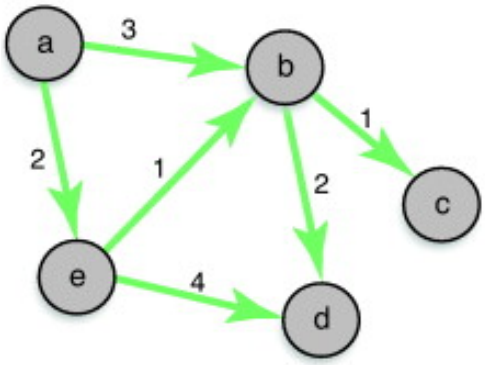
j \ i	a	b	c	d	e
a	0	1	0	0	1
b	1	0	1	1	1
c	0	1	0	0	0
d	0	1	0	0	1
e	1	1	0	1	0

(a) Matrika sosednosti, če je omrežje neuteženo

j \ i	a	b	c	d	e
a	0	3	0	0	2
b	3	0	1	2	1
c	0	1	0	0	0
d	0	2	0	0	4
e	2	1	0	4	0

(b) Matrika sosednosti, če je omrežje uteženo

Slika 2.1: Enostavno neusmerjeno omrežje s petimi vozlišči ter šestimi povezavami in pripadajoče matrike sosednosti



j \ i	a	b	c	d	e
a	0	1	0	0	1
b	0	0	1	1	0
c	0	0	0	0	0
d	0	0	0	0	0
e	0	1	0	1	0

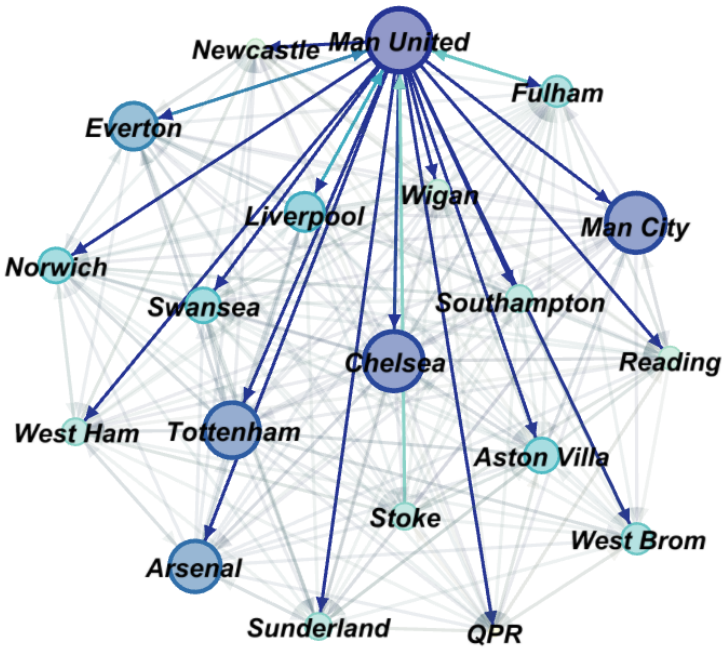
(a) Matrika sosednosti, če je omrežje neuteženo

j \ i	a	b	c	d	e
a	0	3	0	0	2
b	0	0	1	2	0
c	0	0	0	0	0
d	0	0	0	0	0
e	0	1	0	4	0

(b) Matrika sosednosti, če je omrežje uteženo

Slika 2.2: Enostavno usmerjeno omrežje s petimi vozlišči ter šestimi povezavami in pripadajoče matrike sosednosti

Primer kombinacije različnih tipov omrežij navedimo omrežje nogometnih tekem (glej slika 2.3), kjer so vozlišča ekipe, povezave pa odigrane tekme. Omrežje je neuteženo in usmerjeno, kjer vhodne povezave pomenijo poraz ekipe, izhodne pa zmago proti ekipi na katero kaže povezava.



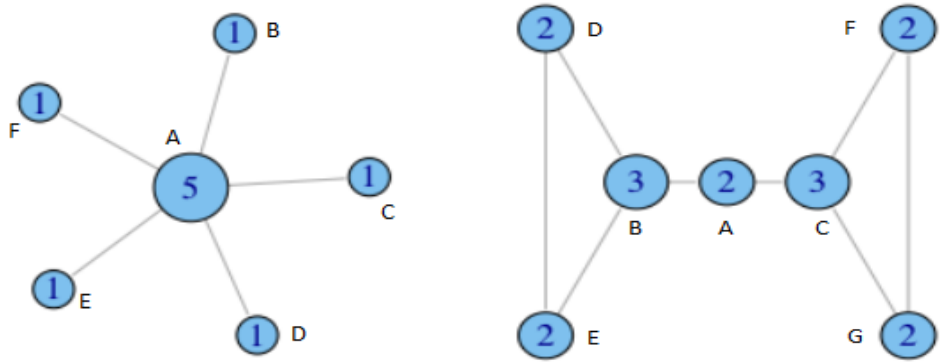
Slika 2.3: Primer kombinacije neuteženega in usmerjenega omrežja. Poudarek je na eno vozlišče, ki predstavlja zmagovalec lige, medtem pa je velikost vozlišča sorazmerna s številom zmag oziroma izhodnih povezav

2.1.2 Mere vozlišč

Če poznamo strukturo omrežja, lahko iz njega izračunamo veliko uporabnih količin oziroma mer, ki zajemejo posebne značilnosti tovrstnega omrežja. Veliko najpomembnejših idej na tem področju prihaja iz družbenih ved, oziroma od discipline analize socialnih omrežij. Kljub temu, se opisane mere sedaj na široko uporabljalo tudi zunaj socialnih omrežij, vključno z računalništvom, fiziko in biologijo ter predstavljajo pomemben del osnove omrežnih orodij. V nadaljevanju sledi pregled mer s poudarkom na tistih, ki so značilne za delo z našim omrežjem.

Velik obseg raziskave na področju omrežij je bila namenjena t.i. koncepta središčnost. Kot smo že povedali v razdelku 2.1.1, razlikujemo usmerjena in neusmerjena omrežja. Pri neusmerjenih omrežjih najpreprostejša središčnost vozlišča predstavlja le število povezav tega vozlišča z ostalimi v omrežju in

se imenuje stopnja središčnosti (*angl. Degree Centrality*). Pri usmerjenih omrežjih pa obstajata vhodna stopnja (angl. in-degree) in izhodna stopnja (angl. out-degree) središčnosti. Vhodna stopnja predstavlja število povezav, ki gredo proti vozlišču, izhodna pa predstavlja število povezav, ki izhajajo iz vozlišča [27]. V našem primeru vhodne povezave pomenijo poraz ekipe, izhodne pa pomenijo zmago proti ekipi na katero kaže povezava.



(a) Omrežje z izstopajočim vozliščem glede vrednosti DC-ja

(b) Omrežje z uravnoteženimi vozlišči glede vredsnoti DC-ja

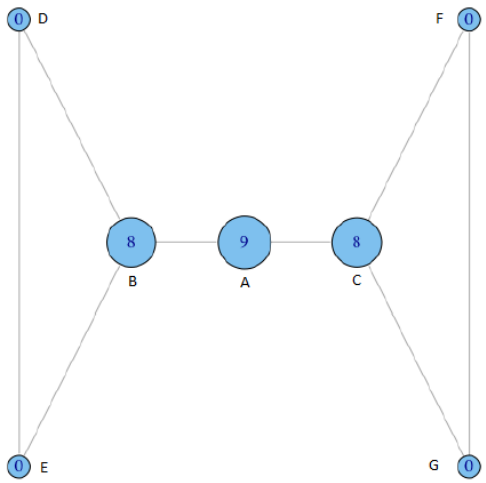
Slika 2.4: Primer dve majhni neusmerjeni omrežji s prikazanimi vrednosti mere DC

Na sliki 2.4 sta predstavljena dva neusmerjena grafa, kjer je za vsako vozlišče izračunana mera DC. S sliko želimo pokazati, da samo število povezav ni dovolj. V primeru 2.4a je najbolj središčno vozlišče A, kar je tudi pričakovano, saj ima največ povezav. Po drugi strani pa, če analiziramo primer 2.4b bomo opazili, da imajo več vozlišč DC=2 in ker je topologija omrežja drugačna , se nam poraja vprašanje: Ali imajo vozlišče A ter vozlišča D, E, F in G enak vpliv na omrežje? Na primer, če vozlišče D želi komunicirati z vozliščem F, mora nujno iti skozi vozlišče A. V tem primeru lahko rečemo, da vozlišče A povezuje dve skupini vozlišč in to naj bi bilo najpomembnejše, vendar v našem primeru to ne drži, saj ima enako pomembnost kot ostala vozlišča z DC=2. Kako to zajamemo? Obstaja pojem, ki se imenuje posre-

dništvo (angl. brokerage). DC je odvisen le od število povezav, posredništvo pa ni podprto. V tem primeru moramo uporabljati drugo mero, ki podpira posredništvo in se imenuje BC.
Središčnost vmesnosti(*angl. Betweenness Centrality*) je koncept središčnosti, njena osnovna ideja pa je predstaviti število najkrajših poti od enih vozlišč do drugih, ki gredo skozi druga vozlišča [2]. Definirana je kot:

$$C_B(i) = \sum_{j<k} g_{jk}(i) \,/\, g_{jk} \ , \tag{2.4}$$

kjer g_{jk} je število najkrajših poti, ki povezujejo vozlišča j in k, $g_{jk}(i)$ pa je število najkrajših poti, v katerih je prisotno vozlišče i.



Slika 2.5: Omrežje s poudarkom na vrednosti BC-ja

Če se vrnemo na prejšnje omrežje (2.4b), središčno vozlišče A, ki je imelo prej stopnjo 2, zdaj ima največjo BC=9 (glej Slika 2.5). Razlog za to gre iskati v upoštevanem posredništvu. Po drugi strani pa imajo vozlišča D, E, F in G, BC=0, ker ne sodijo v najkrajše poti ostalih vozlišč.

V splošnem ima vozlišče z visoko središčnostjo velik vpliv na prenos podatkov preko omrežja, s predpostavko, da prenos sledi najkrajši poti. Edina

slabost BC-ja je naslednje: če vozlišče z visokim BC-jem odstranimo iz omrežja, pretok informacij ne bo več tekel. Na primer, če z omrežja odstranimo vozlišče A, skupina vozlišč B, D in E ne bo mogla več komunicirati s skupino vozlišč C, F in G.

Bližinska središčnost (*angl. Closeness Centrality*) predstavlja povsem drugačna mera središčnosti, ki je definirana kot vsota obratnih vrednosti vseh razdalj iz izbranega vozlišča i do vsakega drugega vozlišča j v omrežju:

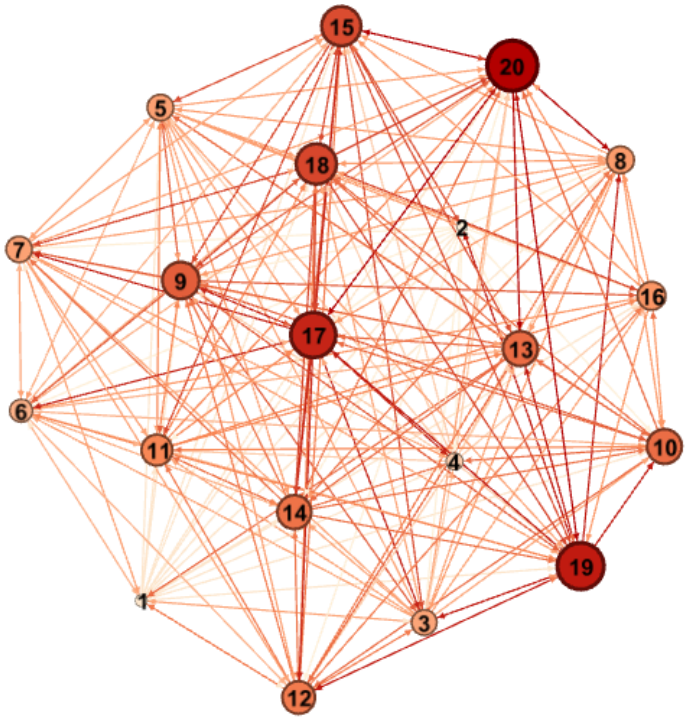
$$C_c(i) = \left[\sum_{j=1}^{N} d(i,j) \right]^{-1}, \tag{2.5}$$

kjer je N število vseh vozlišc v omrežju, i osrednjo vozlišče v omrežju, j je neko drugo vozlišče in $d(i,j)$ najkrajša pot med tema dvema vozliščama. CC izhaja iz tega razloga, da ni pomembno ali informacija, ki se prenaša skozi omrežje gre čez določeno vozlišče, vendar je poudarek bolj na tem, da ima vozlišče enostaven dostop do velikega dela omrežja [13]. V splošnem to pomeni, da vozlišča želijo narediti čim manj korakov do ostalih vozlišč, medtem ko je v našem omrežju mera sorazmerna z uvrščenostjo ekipe. Torej, čim je večja vrednost CC-ja, je ekipa tudi višje uvrščena na končni lestvici.

Naravna razširitev preprostega DC-ja je središčnost lastnega vektorja (*angl. Eigenvector Centrality*), ki meri pomembnost določenega vozlišča v omrežju. Izhaja iz razloga, da vsa vozlišča v omrežju niso enakovredna. EC pokaže koliko je vozlišče središčno, glede na središčnosti njegovih sosedov. Algoritem je rekurziven, saj si ti pomemben toliko, kot so pomembni tvoji sosedi. Tvoji sosedi pa so pomembni toliko, kot so pomembni njihovi sosedi [4]. Definiran je kot:

$$c_i^e = \frac{1}{\lambda} \sum_{j:j \neq i} \omega_{i,j} c_j^e, \tag{2.6}$$

kjer je λ konstanta, $\omega_{i,j}$ utež med vozliščema i in j in c_j^e vrednost EC j-tega vozlišča.

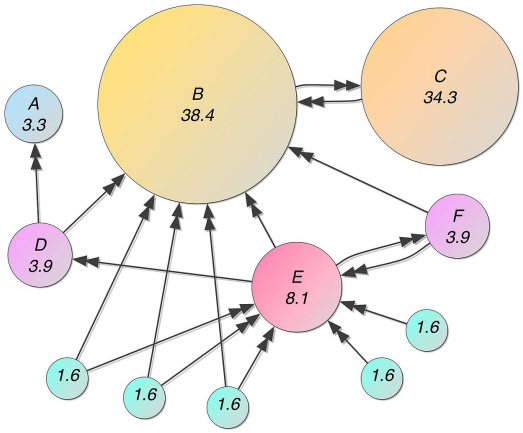


Slika 2.6: Primer omrežja z 20 vozlišči in 226 usmerjenih povezav

Na sliki 2.6 je predstavljeno omrežje z dvajsetimi vozlišči ter 226 povezav. Vozlišča so ekipe, vhodne povezave pomenijo poraz ekipe, izhodne pa zmago proti ekipi, na katere kaže povezava. Na omrežju je prikazan vpliv EC-ja, torej bolj kot je veliko in rdeče obarvano vozlišče, toliko je to pomembnejše. Številke v vozliščih pomenijo končno uvrstitev ekipe po 38 tekmah in kot lahko opazimo nizko uvrstitev ekip, imajo boljšo mero kot visoko uvrščene. Razlog je razviden, saj je zmaga nad ekipo, ki je visoko uvrščena pomembnejša kot zmaga nad ekipo, ki je nizko uvrščena.

Podobna mera kot EC je uvrstitev strani (*angl. PageRank*). PR predstavlja algoritam, ki ga *Google* uporablja za uvrstitev strani v svojih iskalnih algoritmih. Razvil ga je Larry Page, eden od ustanoviteljev *Googla* leta 1996. Algoritem deluje tako, da šteje število kakovostnih povezav na določeno stran,

da bi določil grobo oceno o tem, kako pomembna je ta spletna stran. Osnovna predpostavka je, da bodo verjetno od pomembnejših spletnih strani prejeli več povezav iz drugih spletnih strani [11, 21].

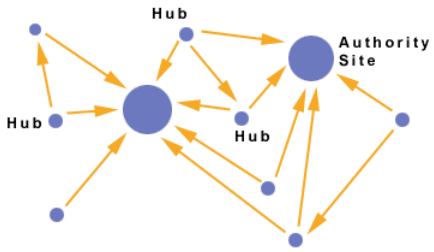


Slika 2.7: Primer omrežja, kjer je razviden vpliv mere PR [29]

.

Kot lahko opazimo na sliki 2.7, ima vozlišče C večjo PR vrednost kot vozlišče E, čeprav ima manj povezav. Razlog za to je, da edina povezava do vozlišča C prihaja iz vozlišča, ki ima visoko PR vrednost in zato je pomembnost večja, dokler je vozlišče E povezano z več drugimi vozlišči, vendar brez večjega pomena.

Z razvojem interneta in povezovanjem spletnih strani, se je pojavila mera kazala in viri (*angl. hubs and authorities*). Ideja mere pomeni, da povezave predstavljajo glasovi in stran je pomembnejša, če ima več povezav. Glede na usmerjenost povezave obstajata dve vrsti vozlišč. Prva so viri, ki vsebujejo koristne informacije za določeno tematiko, druga pa so kazala, ki kažejo, kje lahko najboljše vire najdemo [9].



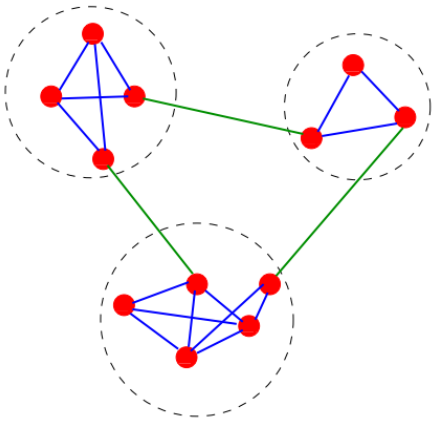
Slika 2.8: Primer majhnega omrežja s poudarkom na meri HUB in AUTH [17].

Vozlišče bo imelo visoko mero AUTH, če ima več vhodnih povezav, oziroma visoko mero HUB, če ima več izhodnih povezav. V našem omrežju vozlišča z visoko mero HUB so ekipe, ki so dosegle veliko zmag, dokler vozlišča z visoko mero AUTH predstavljajo ekipe, ki imajo veliko porazov. Algoritem za računanje mere je HITS.

2.1.3 Odkrivanje skupnosti

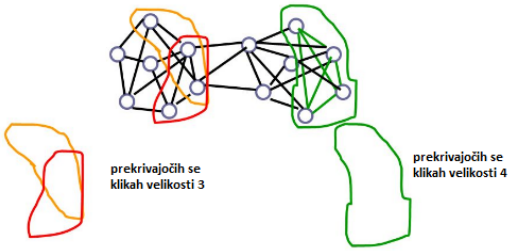
Odkrivanje skupnosti (*angl. Community detection*) oziroma razvrščanje podatkov v skupini, predstavlja ena od najbolj osnovnih tehnik za analiza podatkov. Skupnosti, imenovane tudi grozdi so skupine vozlišč, ki delijo veliko skupnih lastnosti z vozlišči znotraj skupine in zelo malo lastnosti z vozlišči iz drugih skupin [12, 31]. Osnovni algoritem za odkrivanje skupnosti je hierarhično razvrščanje (angl. hierarchical clustering) [25].

Danes je področje zelo popularno v socialnih omrežij, kjer se ljudje delijo v določene skupnosti. Kljub temu, pa je tudi razumljivo, da so ljudje povezani z več člani drugih skupnosti. Na primer oseba ima lahko povezave do različnih družbenih skupin kot so družina, prijatelji, sodelavci itd. V splošnem število skupnosti v katerem spada vozlišče je neomejeno, kajti lahko se poveže s toliko skupinami kot si želi. Tovrstni primer v analizo omrežjih imenujemo prekrivanje skupin (*angl. overlapping communities*) [20, 39]. Za razliko od navadnega iskanja skupnosti, ta metoda temelji na algoritmu per-



Slika 2.9: Preprosto omrežje s tremi skupnostmi, označene s prekinjenimi krogi [12].

kolacija klikov (angl. clique percolation) [8]. Osnovna ideja pa je najti čim več klikov znotraj posamezne skupine.



Slika 2.10: Omrežje, kjer so predstavljeni prekrivajočih se klikah

Na sliki 2.10 opazimo, da se kliki lahko tudi prekrivajo, kar je dodatna prednost algoritma. V našem omrežju smo tovrstno metodo uporabljali za iskanje skupnosti med ekipami.

2.2 Podatkovno rudarjenje

Podatkovno rudarjenje predstavlja interdisciplinarno področje računalniške znanosti. Ponavadi se uporablja tudi izraz odkrivanje znanj iz podatkov, zato ker je postopek analiziranja podatkov iz različnih gledišč ter povzemanje v

uporabne informacije. Tehnično gledano, podatkovno rudarjenje je postopek iskanja korelacij ali vzorcev iz množice podatkov, z uporabo metod iz strojnega učenja, statistike in podatkovnega skladišča. Cilj je razumevanje dobljenih vzorcev, katere bi kasneje uporabili za napoved. Poleg iskanja vzorcev gre lahko tudi za iskanje sprememb ter anomalij.
Za učinkovito podatkovno rudarjenje smo v diplomski nalogi sledili metodologiji CRISP-DM [10, 15, 34]. To je standard, ki se izvaja v več fazah. Faze se izvajajo zaporedno in dokler se ena faza ne konča, se druga ne more začeti. Prva faza je *Razumevanje problema*, kjer spoznamo naš problem in tudi določimo naš cilj. Druga faza je *Razumevanje podatkov*, ki pa temelji na spoznavanju podatkov ter iskanju izjem. Zatem sledi tretja faza *Priprava podatkov*, ki je najpomembnejša in običajno zahteva veliko časa. V tej fazi vrednotimo, poenotimo, počistimo, filtriramo ter transformiramo podatke. Po uspešni pripravi se nadaljuje v četrto fazo, tako imenovano *Modeliranje*, kjer najprej določimo metode, s katerimi bomo rudarili. Nato razdelimo podatke na učno in testno množico ter zgradimo model. Ta faza je močno povezana s prejšno fazo in pogosto se dogaja, da se vrnemo nazaj in spet popravljamo podatke, zato ker zgrajeni model ni bil zgrajen po naših pričakovanjih. Sledi zadnja peta faza *Vrednotenje in uporaba*. V tej fazi vrednotimo dobljene rezultate, ocenimo kako uspešen je bil naš model ter ugotovimo kako, kdo in kdaj bo uporabljal rezultate (odkrito znanje, modele). Zakaj sploh smo uporabljali metodologijo? Zato ker:

- mora biti proces zanesljiv in ponovljiv ;
- nudi pomoč pri načrtovanju in upravljanju podatkovnega rudarjenja ;
- daje vtis stabilnosti in zrelosti področja .

Obstajata dva načina podatkovnega rudarjenja:

Klasifikacija (angl. *classification*) - podatki iz množice se razdelijo v več razredov. Na primer: imamo podatke o številu točk in ekipe, ki imajo več

kot 50 točk, torej so dobro uvrščene, ostale pa so slabo. Razred je disrektna spremenljivka.

Regresija (angl. *regression*) - statistična metoda namenjena preiskovanja ter modeliranju povezanosti med spremenljivkami. Ima široko možnost uporabe na različnih področij. Uporablja se v inženirstvo, fiziki, kemiji, ekonomiji, menagmentu, biotehničnih znanostih in družbenih vedah. Danes predstavlja eno izmed najpogoste uporabljenih statističnih metod.

V diplomski nalogi skušamo ugotoviti, katere spremenljivke so najbolj povezane ter njihov vpliv pri napovedovanju rezultatov, zato uporabljamo različne regresijske metode.

2.2.1 Regresijske metode

V tem podrazdelku so opisane regresijske metode, katere smo uporabili v diplomski nalogi [19].

Linearna regresija (*angl. Linear regression*) je pristop za modeliranje razmerja med skalarno odvisno spremenljivko in eno ali več neodvisnih spremenljivk [24]. Ko imamo eno neodvisno spremenljivko, je to enostavna linearna regresija, ko pa imamo več neodvisnih spremenljivk, je to multipla linearna regresija. Primer, vprašamo se: ali lahko napovemo število točk na podlagi število golov oziroma ali lahko napovemo število točk na podlagi število golov in številom strelov proti golu. Predstavlja najbolj osnovno metodo in se v praksi pogosto uporablja.

Metoda k-najbližjih sosedov (*angl. k-nearest neighbors*) označena je s k-NN. To je metoda, kjer je napoved razdeljena v dve fazi. V prvi fazi (učenja) si model zapomni vse učne primere, ki jih potem v drugi fazi (uvrščanja) uporablja tako, da za dan primer poišče njemu najbolj podobne primere (sosedov) [23]. Pogosto je neuspešna, ker vsak razred zagotavlja veliko možnih prototipov in je posledično odločitvena meja pogosto nepravilna. Pri klasifikaciji k-NN vrne diskretno vrednost, pri regresiji pa zvezno.

Regresijska drevesa (*angl. Regression tree*) so ena najbolj osnovnih in enostavnih metod za gradnjo napovednega modela. Ideja algoritma je razbitje začetne množice podatkov na razrede oziroma čimbolj čiste podmnožice [32]. Napoveduje vrednosti ciljne spremenljivke, ki temelji na že znani učni množici. Regresijska drevesa so podobna kot klasifikacijska, edina razlika je, da mora biti razred zvezen, ostali atributi pa so lahko zvezni ali diskretni.

Naključni gozd (*angl. Random forest*) je metoda, ki deluje z izgradnjo množice odločitvenih dreves v času učenja in jih nato v fazi uvrščanja povpreči. Napove razred, kamor je primer uvrščen, oziroma v katero odločitveno drevo spada napovedan razred [22]. Danes predstavlja ena izmed najbolj uporabnih in učinkovitih metod.

Metoda podpornih vektorjev (*angl. Support vector machine*) deluje tako, da nadzoruje učne primere z njimi povezanih učnih algoritmov, ki analizirajo podatke uporabljene za razvrščanje. Osnovna ideja je najti hiperravnino, ki najbolje razdeli primere nasprotnih razredov [37]. Algoritem je primeren za učenje na velikih množicah primerov, ki so opisani z manj pomembnimi atributi. Danes predstavlja eno od najučinkovitejših metod v strojnem učenju.

Povprečje (*angl. Mean*) je najpreprostejša metoda, ki temelji na povprečne vrednosti učne množice. Deluje tako, da za napovedano vrednost vedno vzame povprečno vrednost nabora razredov, ki jih napovedujemo. V praksi predstavlja metoda z največjim odstopanjem pri napovedovanju, saj se najmanj prilagaja posameznim primerom. Zato služi le kot spodnjo mejo merila uspešnosti za primerjavo z ostalimi algoritmi.

2.2.2 Mere uspešnosti

Za ocenjevanje uspešnosti regresijskih metod smo uporabljali naslednje mere [18]:

Povprečna kvadratna napaka (*angl. Mean squared error*) meri povprečni kvadrat razlike med napovedano in pravo vrednostjo. Predstavljena je s formulo:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2, \tag{2.7}$$

kjer je p_i napovedana vrednost primera i, a_i je prava vrednost primera i in n je število vseh primerov.

Povprečna absolutna napaka (*angl. Mean absolute error*) je definirana kot:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - a_i| = \frac{1}{n}\sum_{i=1}^{n}|e_i| \tag{2.8}$$

Kot že samo ime pove, MAE je povprečje absolutnih napak $|e_i| = |p_i - a_i|$, kjer je p_i napovedana vrednost primera i in a_i prava vrednost primera i.

Relativna kvadratna napaka (*angl. Relative squared error*) je definirana kot:

$$RSE = \frac{\sum_{i=1}^{n}(p_i - a_i)^2}{\sum_{i=1}^{n}(\bar{a} - a_i)^2}, \tag{2.9}$$

kjer je p_i napovedana vrednost primera i, a_i prava vrednost primera i in $\bar{a}$ povprečna vrednost celotnega primera a.

Relativna absolutna napaka (*angl. Relative absolute error*) je definirana kot:

$$RAE = \frac{\sum_{i=1}^{n} |p_i - a_i|}{\sum_{i=1}^{n} |\bar{a} - a_i|} , \tag{2.10}$$

kjer je p_i napovedana vrednost primera i, a_i prava vrednost primera i in $\bar{a}$ povprečna vrednost vseh primerov a-ja.

2.2.3 Mere korelacije

Korelacija predstavlja mero, s katero merimo linearno povezanost dveh spremenljivk. Formalno, odvisnost se nanaša na primer, v katerih slučajne spremenljivke ne zadovoljuje matematični pogoj verjetnostne neodvisnosti. Korelacije so zelo uporabne, saj lahko kažejo na napovedno razmerje, ki ga je mogoče izkoriščati tudi v praksi. V primerjavi z odvisnostjo, kjer imamo vpliv ene spremenljivke na vrednosti druge, pri korelaciji imamo relacijo, kjer se vrednosti obeh spremenljivk spreminjata hkrati. Če definiramo ene spremenljivke z X in druge z Y lahko razlikujemo več vrst korelacij [1, 6]. Glede na smer povezanosti:

- pozitivna: z naraščanjem X narašča tudi Y ;
- negativna: z naraščanjem X se vrednosti Y zmanjšujejo .

Glede na obliko povezanosti:

- linearna - premočrtna ;
- nelinearna – krivuljna (več vrst, krivulje 2, 3 in nadaljnjega reda) .

Glede na stopnjo povezanosti:

- neznatna ;
- šibka ;

- močna .

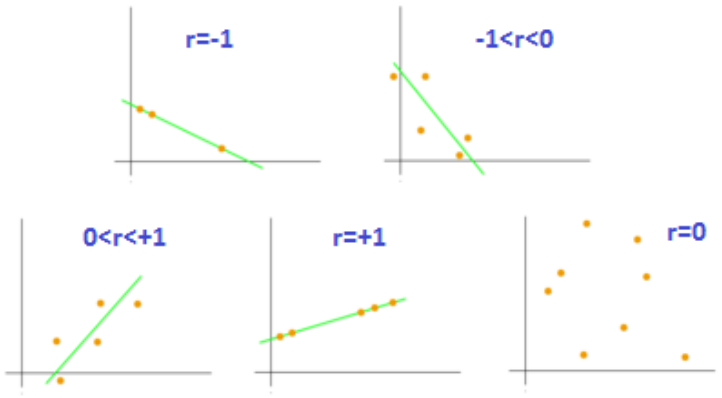
Korelacija je predstavljena s korelacijskimi kvocienti, ki merijo stopnjo (moč) povezanosti ter smer povezave. Danes obstajajo več korelacijskih kvocientov, ki so pogosto označene z ρ ali r. Najpogostejši med njimi je Pearsonov kvocient korelacije [6], ki je občutljiv le na linearno povezanost med dvema spremenljivkama ter nam pove stopnjo in smer povezave. Definiran je kot:

$$r = \frac{\sum(X - \bar{X})(Y - \bar{Y})}{\sqrt{\sum(X - \bar{X})^2}\sqrt{\sum(Y - \bar{Y})^2}} , \tag{2.11}$$

kjer z X so predstavljene vrednosti prve spremenljivke, z Y vrednosti druge spremenljivke, z $\bar{X}$ povprečno vrednost vseh primerov X in z $\bar{Y}$ povprečno vrednost vseh primerov Y.
Glede stopnje povezanosti, vrednost kvocienta se nahaja v intervalu [-1, 1]. Izračunano vrednost se lahko predstavi tudi tako:

- ± 0,00 - ni povezanosti ;
- ± 0,01-0,19 - neznatna povezanost ;
- ± 0,20-0,39 - nizka/šibka povezanost ;
- ± 0,40-0,69 - srednja/zmerna povezanost ;
- ± 0,70-0,89 - visoka/močna povezanost ;
- ± 0,90-0,99 - zelo visoka/zelo močna povezanost ;
- ± 1,00 - popolna (funkcijska) povezanost .

Na sliki 2.11 je prikazano kako se obnaša linearna premica glede vrednosti Pearsonovega kvocienta korelacije.

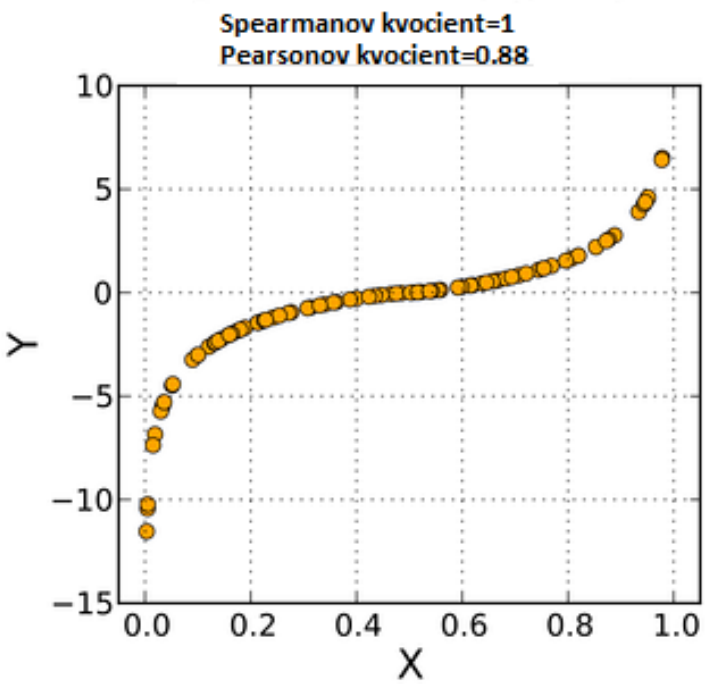


Slika 2.11: Primer vrednosti za Pearsonovega kvocienta korelacije [30].

Če spremenljivki X in Y ne zadoščata pogojem za uporabo Pearsonovega kvocienta se uporablja t.i. Spearmonov kvocient korelacije [6]. V bistvu, Spearmanov kvocient predstavlja izpeljanka Pearsonovega kvocienta, kjer se podatki preoblikujejo v range. Dodatna prednost je tudi to, da je lahko povezanost spremenljivk nelinearna. Na primer, pri kvadratni porezdelitvi bi imel Pearsonov kvocient vrednosti okrog 0, torej bi kazal, da ni povezanosti, čeprav povezava med spremenljivkami obstaja. Spearmanov kvocient se izračuna s formulo:

$$r_s = 1 - \frac{6 \sum D^2}{n(n^2 - 1)} \ , \tag{2.12}$$

kjer je n število vseh parov ranga in d_i razlika rangov med paroma vrednosti spremenljivke X in Y. Pomen vrednosti kvocienta je isti kot pri Pearsonovem kvocientu.



Slika 2.12: Primer, kjer se vidi izboljšava Spearmanovega kvocienta korelacije [35].

Poglavje 3

Orodja in tehnologije

Pri izboru orodij in tehnologij smo največ pozornosti namenili knjižnici za grajenje omrežja. Po pregledu več orodij smo se odločili za vmesnika SNAP ter programski jezik Python.

SNAP [36] je splošno namenski, visoko zmogljiv sistem za analize in manipulacije z omrežji. Razlog zakaj smo se odločili za SNAP je naslednji, omogoča hitro gradnjo omrežja z veliko vozlišč in povezav. Glavna prednost je še ta, da ima že implementirano veliko metod za izračun mer, ki jih potrebujemo za analizo in omogoča hitro gradnjo usmerjenih in neusmerjenih grafov. Ko je omrežje zgrajeno, se nam ponudi številne možnosti za enostavno shranjevanje omrežja v več različnih oblik. Omenjeni odločitvi je botrovalo tudi dejstvo, da je knjižnico Snap.py moč uporabljati v programskem jeziku Python [33].

Slednjega smo privzeli kot našo glavno okolje za delo s podatki, zato ker je enostaven, razmeroma hiter, ima dinamične podatkovne tipe in veliko knjižnic, ki imajo že implementirane metode, katere bomo potrebovali tekom diplomske naloge. Danes je eden med najbolj priljubljenimi programskimi jeziki, predvsem pri delu z velikimi količinami podatkov. Sklada se tudi z orodjem Orange Canvas, ki smo ga uporabljali za gradnjo napovednega mo-

dela.

Orange Canvas je orodje, ki ga je razvilo in vzdržuje Laboratorij za bioinformatiko na Fakulteti za računalništvo in informatiko Univerze v Ljubljani [28]. Uporaba le- tega je možna na več platformah, kot so Microsfot Windows, Linux ter Mac OS X. Gre za odprtokodno orodje za podatkovno rudarjenje ter analizo in vizualizacijo podatkov, tako za začetnike, kot tudi za strokovnjake. Orange Canvas ponuja dva načina obdelave podatkov: preko vmesnika (vizualno programiranje) ali z uporabo programske knjižnice v Pythonu (skriptni Python). Vmesnik ponuja veliko gradnikov, s katerimi se lahko naredi več kot s skriptnim delom. Seveda skriptni Orange ponuja podobne možnosti, ampak njegova uporaba je bolj za grajenje modela, ki vsebuje veliko podatkov.
V diplomski nalogi smo večinoma uporabljali vmesnik, ki je podpora za analizo dobljenih lastnosti iz analize omrežij ter ocenjevanje uspešnosti in prednosti atributov za gradnjo napovednega modela. Poleg tega je nam zelo koristila njegova vizualizacija podatkov, s čimer smo lažje in hitreje analizirali podatke ter primerjali rezultate.

Za delo z omrežjem smo uporabljali tudi orodje Gephi [14]. To je orodje za interaktivno vizualizacijo in predstavlja raziskovalno platformo za vse vrste omrežij, kompleksnih sistemov, dinamičnih in hierarhičnih grafov. Deluje na Microsoft Windows, Linux ter Mac OS X in je odprtokodno orodje. Za Gephi smo se odločili zato, ker ponuja:

- uvoz in shranjevanje omrežja v več oblik ;
- upravljanje z uvozenimi podatki (dodajanje, brisanje, spreminjanje, kopiranje) ;
- metode (algoritmi) za analizo omrežij ;
- osnovna in napredna vizualizacija omrežja ;

- statistična obdelava in filtriranje podatkov ;
- veliko postavitve omrežja ;
- upravljanje z omrežjem (spreminjanje barv, velikosti ter imena povezav in vozlišč).

Za potrebe iskanja skupin smo uporabljali orodje CFinder [5]. To je brezplačno orodje, ki je razvito v programskem jeziku Java. Orodje se uporablja izključno za iskanje in vizualizacijo navadnih oziroma prekrivajočih skupinah vozlišč v omrežju, ki temeljijo na prekrivajočih se klikah (t.i. polni podgraf) v omrežju.

Poglavje 4

Rezultati in interpretacija

V razdelku 2.1 smo spoznali veliko pristopov oziroma metod in tehnik za analizo omrežij. Ti pristopi so nam pomagali čim natančneje analizirati tekme in so bili tudi odlična podlaga pri grajenju napovednega modela. V tem razdelku je najprej podrobneje opisana množica vhodnih podatkov, zatem pa sledi krajša analiza metode odkrivanje skupnosti. Kasneje je opisan eden izmed pomembnejših delov naloge, kjer so predstavljeni rezultati oziroma uspešnost napovednega modela.

4.1 Podatki nogometnih tekem

Podatke o nogometnih tekmah smo pridobili na spletni strani www.football-data.co.uk, kjer so na voljo vsi podatki o tekmah za različne lige (angleška, nemška, španska, francoska, italijanska, nizozemska itd.) v obdobju od sezone 1993/1994, do danes. V diplomski nalogi smo se osredotočili na angleško ligo, ker je to ena izmed najbolj gledanih, obiskovanih in zanimivih lig na svetu [7].

Podatki so prosto dostopni v .csv obliki in je vsaka sezona predstavljena v posebni datoteki. Datoteka je sestavljena iz toliko vrstic, kolikor tekem ima ena sezona. Vsaka vrstica predstavlja eno tekmo in vsebuje različne statistične podatke o tekmi, kot tudi raznovrstne kvote iz različnih stavnic. Za

naše analize in napovedi smo potrebovali zgolj statistične podatke o tekmi in zato smo kvote iz stavnic odstranili.

Div	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST	AST	HF	AF	HC	AC	HY	AY	HR	AR
E0	17-08-13	Arsenal	Aston Villa	1	3	A	1	1	D	16	9	4	4	15	18	4	3	4	5	1	0
E0	17-08-13	Liverpool	Stoke	1	0	H	1	0	H	26	10	11	4	11	11	12	6	1	1	0	0
E0	17-08-13	Norwich	Everton	2	2	D	0	0	D	8	19	2	6	13	10	6	8	2	0	0	0
E0	17-08-13	Sunderland	Fulham	0	1	A	0	0	D	20	5	3	1	14	14	6	1	0	3	0	0
E0	17-08-13	Swansea	Man United	1	4	A	0	2	A	17	15	6	7	13	10	7	4	1	3	0	0
E0	17-08-13	West Brom	Southampton	0	1	A	0	0	D	11	7	1	2	14	24	4	8	4	0	0	0
E0	17-08-13	West Ham	Cardiff	2	0	H	1	0	H	18	12	4	1	10	7	4	3	0	1	0	0
E0	18-08-13	Chelsea	Hull	2	0	H	2	0	H	22	7	5	2	7	16	5	1	0	1	0	0
E0	18-08-13	Crystal Palace	Tottenham	0	1	A	0	0	D	5	17	3	2	6	9	3	7	1	0	0	0
E0	19-08-13	Man City	Newcastle	4	0	H	2	0	H	20	5	11	1	9	7	8	1	2	3	0	1
E0	21-08-13	Chelsea	Aston Villa	2	1	H	1	1	D	15	7	3	3	12	13	1	2	1	4	0	0
E0	24-08-13	Aston Villa	Liverpool	0	1	A	0	1	A	17	5	3	1	9	8	8	2	3	3	0	0
E0	24-08-13	Everton	West Brom	0	0	D	0	0	D	22	7	8	2	14	15	11	1	1	1	0	0
E0	24-08-13	Fulham	Arsenal	1	3	A	0	2	A	16	18	7	7	10	8	1	8	2	2	0	0
E0	24-08-13	Hull	Norwich	1	0	H	1	0	H	6	13	1	4	14	19	1	5	1	1	1	0
E0	24-08-13	Newcastle	West Ham	0	0	D	0	0	D	16	6	0	1	12	14	5	4	0	1	0	0
E0	24-08-13	Southampton	Sunderland	1	1	D	0	1	A	17	8	6	3	12	13	5	2	2	2	0	0
E0	24-08-13	Stoke	Crystal Palace	2	1	H	0	1	A	14	14	5	5	13	6	7	0	3	0	0	0
E0	25-08-13	Cardiff	Man City	3	2	H	0	0	D	9	16	6	5	2	9	3	8	0	0	0	0
E0	25-08-13	Tottenham	Swansea	1	0	H	0	0	D	19	7	5	5	10	15	8	3	1	4	0	0
E0	26-08-13	Man United	Chelsea	0	0	D	0	0	D	13	8	3	4	9	9	4	1	0	2	0	0

Slika 4.1: Primer podatkov iz datoteke

Na sliki 4.1 je prikazan primer podatkov. Kot lahko opazimo v prvi vrstici se nahajajo kratice s pomenom prikazan v tabeli 4.1.

Kratica	**Pomen**
FTHG	Število golov domače ekipe ob koncu tekme
FTAG	Število golov gostujoče ekipe ob koncu tekme
FTR	Izid tekme ob koncu
HTHG	Število golov domače ekipe ob polčasu
HTAG	Število golov gostujoče ekipe ob polčasu
HTR	Izid tekme ob polčasu
HS	Število strelov domače ekipe
AS	Število strelov gostujoče ekipe
HST	Število strelov proti golu domače ekipe
AST	Število strelov proti golu gostujoče ekipe

HF	Število prekrškov domače ekipe
AF	Število prekrškov gostujoče ekipe
HC	Število kotov domače ekipe
AC	Število kotov gostujoče ekipe
HY	Število rumenih kartonov domače ekipe
AY	Število rumenih kartonov gostujoče ekipe
HR	Število rdečih kartonov domače ekipe
AR	Število rdečih kartonov gostujoče ekipe

Tabela 4.1: Legenda kratic

Moramo omeniti, da vse datoteke niso popolne, oziroma ne vsebujejo vseh podatkov, ki jih potrebujemo. Zato so vse nadaljne analize in napovedi narejene za petnajst- letno obdobje od sezone 2000/2001 do 2014/2015.
Ker smo se osredotočili le na angleško ligo, bomo na kratko pojasnili sistem tekmovanja:

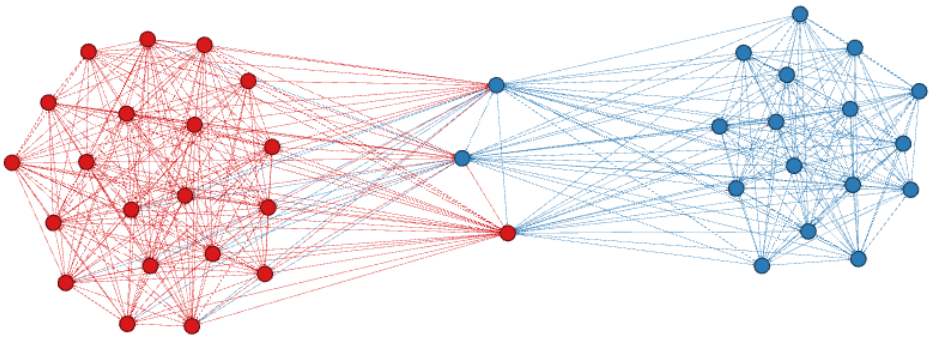
- liga je sestavljena iz 20 ekip ;
- vsaka ekipa igra z vsako ekipo doma in v gosteh, torej imamo 38 krogov in 380 tekem ;
- zmagovalec tekme, oziroma tista ekipa, ki zadene več golov prejme 3 točke, poraženec 0 točk, medtem pa, če se tekma konča z nakim številom golov pri obeh ekipah, obe ekipi dobita po 1 točko ;
- ekipa, ki ima na koncu 38 kroga največje število točk je prvak in se neposredno uvrsti v Ligo prvakov ;
- ekipi, ki končata na 2. in 3. mestu se tudi neposredno uvrstita v Ligo prvakov, ekipa na 4. mestu mora igrati dodatne kvalifikacije ;
- ekipe, ki končajo na 5., 6. in 7. mestu se neposredno uvrstijo v manj pomembnejše tekmovanje v Evropsko ligo ;

- ekipe, ki končajo na 18., 19. in 20. mestu izpadejo iz lige in naslednjo sezono igrajo v drugi angleški ligi (en rang nižje), dokler njihova mesta v prvi ligi nadomestijo ekipe, ki končajo na prvih treh mestih v drugi ligi v tej sezoni ;

- ekipa, ki na koncu sezone prejme najmanjše število kartonov, dobi posebno povabilo iz UEFA za kvalifikacije za Evropsko ligo .

4.2 Odkrivanje skupnosti nogometnih ekip

Kot že vemo iz podrazdelka 2.1.3 je metoda odkrivanja skupnosti primerna le za večja omrežja. Ker pa naše omrežje spada med majhna omrežja, smo sprejeli izziv analizirati dve različni ligi iz iste države, s čimer bi pokazali delovanje tovrstne metode. Torej, omrežje je bilo zgrajeno iz dveh datotek: prva je predstavljala podatke o prvi angleški ligi za sezono 2012/2013, druga pa je vsebovala podatke o drugi angleški ligi za sezono 2011/2012. Iz sistema tekmovanja vemo, da se zadnje tri ekipe iz prve angleške lige preselijo v nižji rang tekmovanja, medtem ko prve tri uvrščene ekipe iz druge lige napredujejo v en rang višje oziroma v prvo ligo.



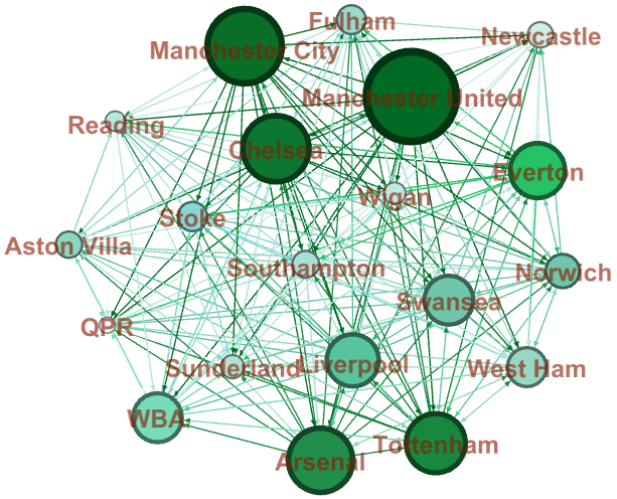
Slika 4.2: Primer omrežja s prekrivajočimi skupinami

Na sliki 4.2 rdeče pobarvana vozlišča predstavljajo ekipe iz druge angleške lige (sezona 2011/2012), modro pobarvana vozlišča pa ekipe iz prve

lige (sezona 2012/2013), medtem ko so trije vozlišči na sredini omrežja ekipe, ki imajo povezavo tako do rdeče kot tudi modre skupine. Pravzaprav so to ekipe, ki so iz druge angleške lige v sezoni 2011/2012 napredovale v prvo angleško ligo in v njej igrali v sezoni 2012/2013. Kot lahko opazimo, algoritem je razvrstil dve vozlišči v modro skupnost in eno v rdečo, vendar vsa tri vozlišča spadajo v obe skupnosti. S tem smo pokazali, da algoritem uspešno deluje na razširjenem omrežju, vendar zaradi kompleksnosti omrežja, nadaljne analize in napovedi nismo delali.

4.3 Napovedovanje uspešnosti nogometnih ekip

Iz množice vhodnih podatkov smo najprej zgradili omrežje, nato smo z merami, ki smo jih dobili od analize omrežij ustvarili datoteko z lastnostmi, katero smo uvozili v Orangu. Lastnosti so naslednje: BC, CC, EC, PR, AUTH in HUB, katere smo podrobneje opisali v 2.1.2. V Orangu smo zgradili napovedni model in poskušali čim natančneje napovedati: število točk, golov, strelov proti golu, kotov, kartonov (število rumenih + število rdečih kartonov) in prekrškov.



Slika 4.3: Omrežje za sezono 2012/2013

Slika 4.3 prikazuje primer omrežja za eno sezono. Vozlišča predstavljajo ekipe, povezave pa tekme med ekipami. Kot lahko vidimo, so povezave usmerjene, kar pomeni, da ekipa v katero je puščica usmerjena je zgubila proti ekipi iz katere izhaja puščica. Barva in velikost vozlišč pomenita, da čim večje in bolj barvno je vozlišče, toliko je ekipa višje na lestvici. V tej sezoni je prvak postal Manchester United a iz lige pa so izpadli Wigan, Reading in QPR, kar se da razbrati tudi iz omrežja.
Zaradi bližine atributov smo se odločili izračunati korelacijo med atributami s pomočjo Pearsonovega in Spearmanovega korelacijska kvocienta.

Tabeli izračunanega povprečnega Pearsonovega in Spearmanovega kvocienta za vseh 15 sezon:

Pearson	Točke	Goli	Streli	Koti	Kartoni	Prekrški
Točke	**1,0000**	**0,9052**	0,7937	0,6339	-0,1947	-0,3249
Goli	**0,9052**	**1,0000**	0,8105	0,6278	-0,2168	-0,3406
Streli	0,7937	0,8105	1,0000	0,7165	-0,2046	-0,3556
Koti	0,6339	0,6278	0,7165	1,0000	-0,1533	-0,2496
Kartoni	-0,1947	-0,2168	-0,2046	-0,1533	**1,0000**	**0,5176**
Prekrški	-0,3249	-0,3406	-0,3556	-0,2496	**0,5176**	**1,0000**

Tabela 4.2: Koreliranost atributov glede Pearsonovega kvocienta

Spearman	Točke	Goli	Streli	Koti	Kartoni	Prekrški
Točke	**1,0000**	**0,8438**	0,7175	0,6022	-0,2318	-0,3017
Goli	**0,8438**	**1,0000**	0,7622	0,5903	-0,2334	-0,3226
Streli	0,7175	0,7622	1,0000	0,6783	-0,2054	-0,2973
Koti	0,6022	0,5903	0,6783	1,0000	-0,1624	-0,2321
Kartoni	-0,2318	-0,2334	-0,2054	-0,1624	**1,0000**	**0,4947**
Prekrški	-0,3017	-0,3226	-0,2973	-0,2321	**0,4947**	**1,0000**

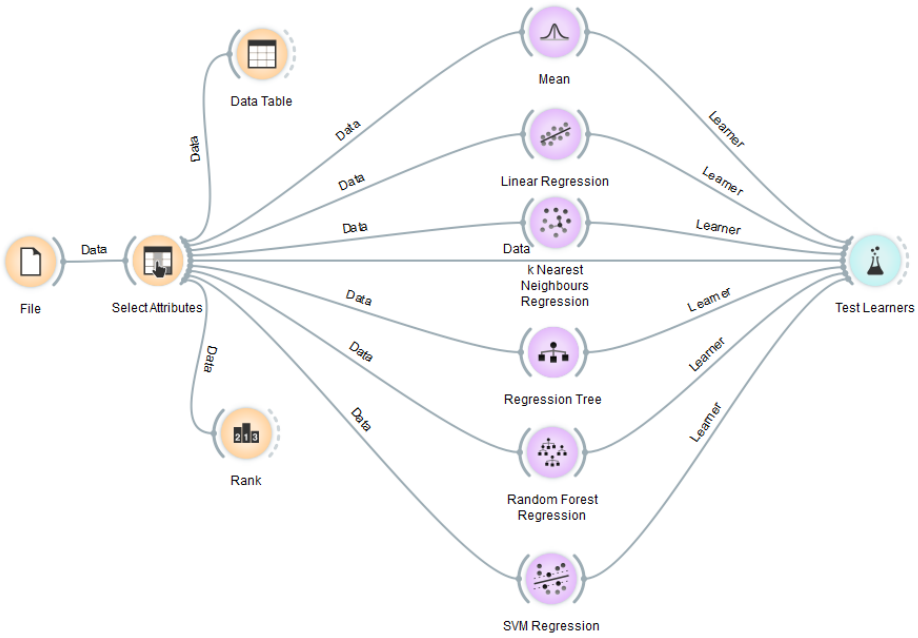
Tabela 4.3: Koreliranost atributov glede Spearmanovega kvocienta

Z opisa tabel lahko sklepamo, da so nekateri atributi zelo močno korelirani med seboj. Pri napovedovanju ne bomo napovedali vseh atributov, ampak bomo le glede na koreliranosti določili skupine.

1. skupina : točke in goli ;
2. skupina : kartoni in prekrški ;
3. skupina : streli proti golu ;
4. skupina : koti .

Prvi dve skupini sta sestavljena iz dveh atributov, ostali dve pa le iz enega. Razlog za to je koreliranost atributov. Kot lahko vidimo v tabelah, imata atributa: streli proti golu in koti najmočnejši koreliran atribut, ki ima močnejšo korelacijo z drugimi atributi. Posledica tega je enojni atribut v skupini.
S koreliranostjo smo dosegli, da smo poenostavili naš napovedni model tako, da ne bomo napovedali tako točke kot tudi gole, ampak le en atribut iz vsake skupine. V našem primeru smo se zaradi pomembnosti atributov odločili, da bodo točke predstavnik 1. skupine in kartoni predstavnik 2. skupine. Ker sta 3. in 4. skupina sestavljena le iz enega atributa, posledično tista dva atributa postaneta predstavnika skupine.
V naslednjem koraku smo zaradi časovne zahtevnosti algoritmov, ki smo jih opisali v razdelku 2.2.1 poskušali ugotoviti, katere lastnosti najbolje napovedo predstavnike skupin. V tej fazi nam je zelo pomagal tudi Orange, ki je pospešil iskanje z njegovim gradnikom Rank, ki nam je s pomočjo algoritma ReliefF povedal pomembnost lastnosti.

Ko smo bili pripravljeni za gradnjo napovednega modela, smo najprej združili vse datoteke s petnajstih sezon v eno in začeli s testiranjem posamičnih lastnosti. Nato smo postopoma iskali kombinacije posamičnih lastnosti z drugimi lastnosti, ki so dajali najboljše rezulate. Zaradi enostavnosti je bilo testiranje narejeno le takrat, ko je bil ciljni napovedani atribut



Slika 4.4: Shema celotne analize v Orangu

število točk. Prav tako je bila zaradi manjše množice vhodnih podatkov pri vzorčenju uporabljena metoda izloči enega (angl. Leave one out), kar pomeni, da se za vsak razpoložljivi primer zgradi eno hipotezo. Nato se primer izloči iz učne množice in se zgradi hipoteza z vsemi ostalimi primeri, katere se potem uporabi v testni množici za reševanje izločenega primera. Za vse primere se to ponovi in dobiš povprečno uspešnost zgrajenih hipotez na izločenih primerih. Rezultati so prikazani v nadaljevanju.

Lastnost/Algoritem	LR	k-NN	SVM	RT	RF	Mean
BC	0,9526	1,1223	0,9159	1,0042	1,0064	1,0042
CC	0,9594	0,7355	0,6839	1,0042	0,6872	1,0042
EC	0,6994	0,8404	0,6453	1,0042	0,7407	1,0042
PR	0,7104	0,8479	0,6636	1,0289	0,7373	1,0042
HUB	0,6633	0,7806	0,6146	1,0859	0,6780	1,0042
AUTH	0,6409	0,7129	0,6038	0,6719	0,6438	1,0042

Tabela 4.4: Mera uspešnosti RAE pri posamičnih lastnostih

Lastnost/Algoritem	LR	k-NN	SVM	RT	RF	Mean
BC, HUB, AUTH	0,6097	0,6957	0,5793	0,7482	0,6150	1,0042
CC, HUB, AUTH	0,6123	0,706	0,5878	0,7482	0,6085	1,0042
EC, HUB, AUTH	0,6109	0,7152	0,5897	0,7536	0,6271	1,0042
PR, HUB	0,6248	0,773	0,6001	1,0859	0,6595	1,0042
vsi	0,6116	0.6790	0,6102	0,7536	0,6150	1,0042

Tabela 4.5: Mera uspešnosti RAE pri kombinaciji lastnostih

Najbolj izrazita lastnost v zgornjih tabelah je BC, ki je kot posameznica imela dokaj visoke napake, dokler se je s kombiniranjem s HUB-om in AUTH-om napak pri vseh algoritmih izdatno izboljšala. Kot lahko opazimo v Tabeli 4.5 kombinacija BC, HUB in AUTH predstavlja najučinkovitejšo kombinacijo lastnosti v primerjavi z ostalimi kombinacijami. Ker je omenjena kombinacija boljša tudi od kombinacije vseh atributov smo se odločili, da bomo nadaljne analize delali le s to kombinacijo lastnosti.

Pri grajenju napovednega modela smo se odločili, da bomo zgradili več vrst omrežij. Vsa omrežja so bila usmerjena in neutežena, razen eno omrežje, kjer smo poskušali z utežmi. Tovrstno omrežje je bilo zgrajeno za obdobje petnajstih sezon, uteži pa so bile predstavljene z relacijo $\frac{1}{n}$, kjer je n število zmag proti ekipi, na katero kaže usmerjena povezava. S takšno vrsto omrežja smo dosegli majhno izboljšavo lastnosti PR. Vendar, kot že vemo iz podraz-

delka 2.1.2 uteži vplivajo le na mero PR, smo hitro ugotovili, da majhna izboljšava te lastnosti ne bo izboljšala naše rezultate, saj smo videli v Tabeli 4.5, da je PR najslabša lastnost za napovedovanje atributov. Zaradi tega smo v nadaljevanju zgradili vrsto omrežja, usmerjeno in neuteženo.
Iz te vrste omrežja smo zgradili štiri različna omrežja, ki so se razlikovala v obdobju vhodnih podatkov. Za prvo omrežje smo vzeli obdobje iz prve sezone, za drugo iz dve sezoni, za tretjo iz tri sezone in za četrto iz štiri sezone. Torej smo imeli 14 omrežij za prvo sezono, npr: zgradili smo omrežje za sezono 2000/2001 in poiskušali napovedati za sezono 2001/2002, zgradili omrežje za 2012/2013 in napovedati za 2013/2014. V primeru, ko se je model učil na podatkih oziroma omrežju iz dveh sezon, smo imeli 13 omrežij, npr: omrežje je zgrajeno na podlagi podatkov iz sezon 2003/2004 in 2004/2005 in napovedni model poskuša napovedati za 2005/2006. Podobno je bilo tudi pri omrežju za triletno učno obdobje modela, kjer smo imeli 12 omrežij, npr: imamo omrežje za sezone 2002/2003, 2003/2004 in 2004/2005, napovedni model pa napoveduje za 2005/2006. Prav tako je bila ista zgodba pri četrtem omrežju, kjer smo imeli 11 omrežij, npr: omrežje je zgrajeno na podlagi podatkov iz sezon 2009/2010, 2010/2011, 2011/2012, 2012/2013, napovedni model napoveduje za 2013/2014. Pri vseh omrežij smo uspešnost napovedovanja merili z MAE in RAE.
Najprej smo naredili napovedi, ko se je napovedni model učil na omrežju zgrajenem iz ene sezone. Da bi vedeli, koliko naš model uspešno zmanjša napake vedno primerjamo z algoritmom Mean, ki služi kot spodnjo mejo merila uspešnosti, saj se sam algoritem najmanj prilagaja posameznim primerom. Pri napovedovanju število kartonov je vrednost RAE povsod nad 1, oziroma nad vrednost Meana, kar pomeni, da je napovedovanje število kartonov neuporabno v praksi.

Algoritem/Napaka	MAE	RAE
LR	8,9400	1,1465
k-NN	9,0999	1,1719

SVM	11,3554	1,4745
RT	9,8284	1,2531
RF	8,3139	1,0614
Mean	8,3397	1,0625

Tabela 4.6: Napaki MAE in RAE pri napovedovanju predstavnik 4. skupine število kartonov - narejeno je povprečje napak za vseh 14 omrežij

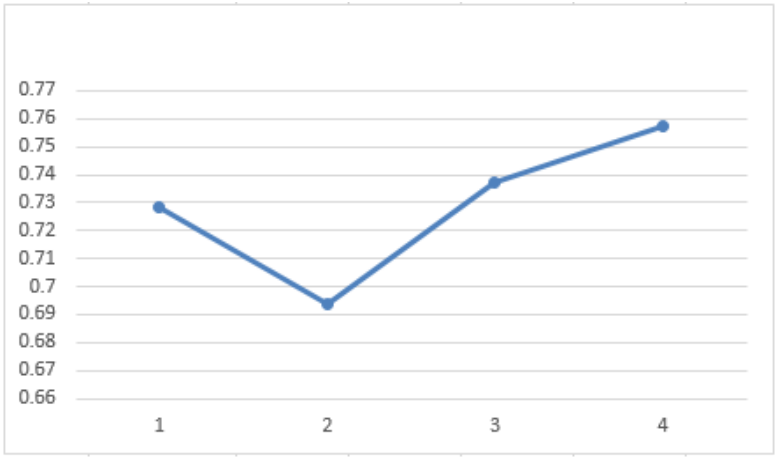
Zaradi zelo slabih rezultatov smo se odločili, da v nadaljevanju omenjeni atribut ne napovedujemo. Pri poskusu napovedovanja število točk, strelov proti golu in kotov si ogledamo kako se mere spreminjajo glede na to, v katerem obdobju je zgrajeno omrežje. Zaradi časovne zahtevnosti smo v tej fazi glede na prvotno dobljene rezultate (vsi rezultati za mere uspešnosti so predstavljene kot povprečje vseh omrežij za določeno obdobje. Kot znano, imamo 14 omrežij za eno- letno obdodje, 13 za dvo- letno, 12 za tri- letno in 11 omrežij za štiri- letno obdobje) še enkrat poenostavili naš model.
Napovedali smo število točk za vsa štiri obdobja:

Algoritem/Obdobje	**MAE**	**RAE**
1 letno obdobje		
LR	10,1229	0,7284
k-NN	10,8058	0,7773
SVM	11,2774	0,8063
RT	12,1530	0,8699
RF	11,7295	0,8402
Mean	14,8471	1,0625
2 letno obdobje		
LR	9,4230	0,6938
k-NN	10,7070	0,7947
SVM	12,2035	0,9035

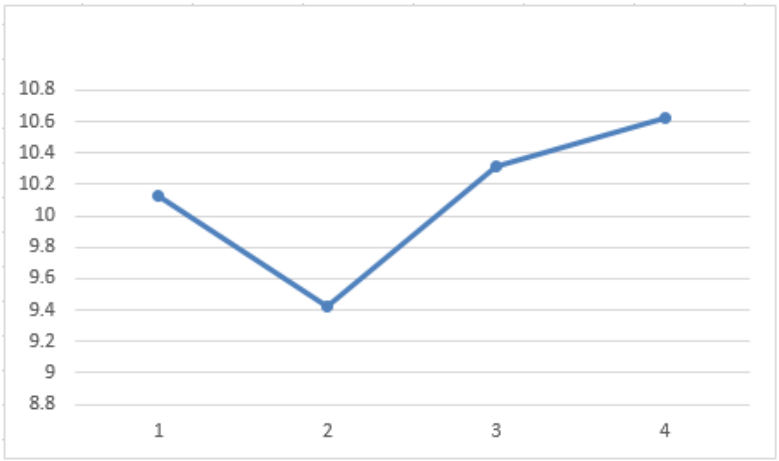
RT	10,4629	0,7792
RF	11,5582	0,8589
Mean	14,5505	1,0603
3 letno obdobje		
LR	10,3137	0,7369
k-NN	11,1050	0,7955
SVM	11,5171	0,8238
RT	11,3655	0,8160
RF	11,8278	0,8477
Mean	14,8441	1,0592
4 letno obdobje		
LR	10,6178	0,7572
k-NN	11,7163	0,8385
SVM	14,5883	1,0339
RT	12,5321	0,8968
RF	12,9093	0,9160
Mean	14,8912	1,0589

Tabela 4.7: Napaki MAE in RAE pri napovedovanju število tock za vsa obdobja

Iz tabele 4.7 je razvidno, da algoritem LR zelo izstopa oziroma izdatno zmanjša napake v primerjavi z ostalimi algoritmi. Zaradi tega smo nadaljne napovedi delali le z omenjenim algoritmom.



(a) RAE



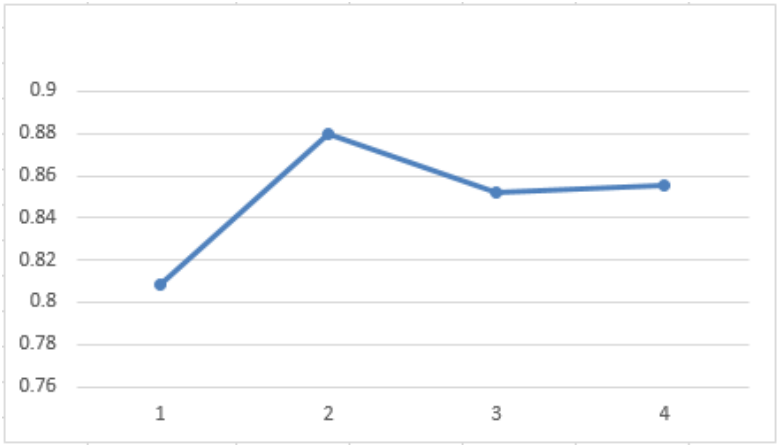
(b) MAE

Slika 4.5: Primerjava napak za različna obdobja pri napovedovanju število točk

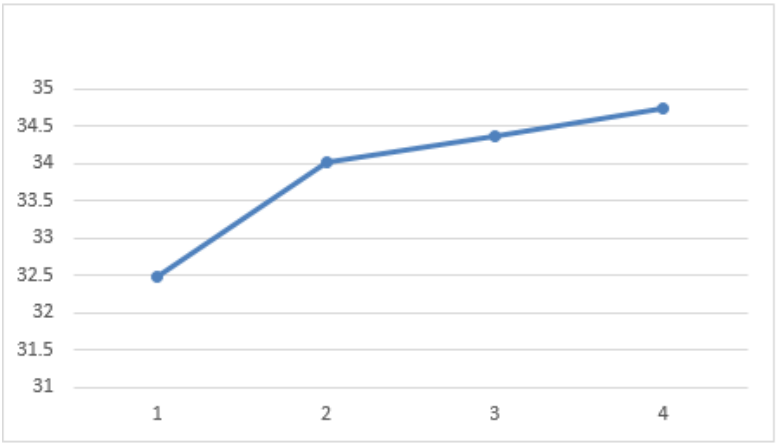
Na Sliki 4.5 so predstavljene napake v obliki grafa. Na osi x se nahaja obdobje iz katerega je omrežje zgrajeno, medtem ko je na osi y predstavljena uspešnost napake. Prav tako lahko opazimo, da dobimo najboljše napovedi oziroma najnižje napake, ko se model uči na dve- letnem obdobju. Pri napaki RAE to pomeni, da se naš model zmoti v slabih 70% in pravilno napove v dobrih 30%. Pri merjenju napake MAE to pomeni, da se model v povprečju zmoti za 9,4 točk. Kot je znano vsaka zmaga prinese 3 točke in če število

točk pretvorimo v število tekem, naš model teoretično v povprečju napačno napove 3 od 38 tekem za vsako ekipo. Čeprav je napaka RAE kar visoka, z MAE smo dobili na videz boljši rezultat, če seveda napako primerjamo s povprečnim številom točk vseh ekip za eno sezono, kar znaša 53,1 (povprečje iz vseh 15 sezon).

Naslednja možna napoved je bilo število strelov proti golu. Prav tako smo v tem primeru kot pri napovedovanju točk, upoštevali le algoritem LR in kombinacijo lastnosti BC, HUB, AUTH. Število strelov proti golu lahko najbolje napovemo, ko se model uči za eno- letno obdobje. Napoved na 32,5 strelov natačno v primerjavi s povprečnim številom strelov v sezoni 214,6 na prvi pogled izgleda v redu, ampak se model zmoti v dobrih 80%. Razlog za slab rezultat gre iskati v množici vhodnih podatkov. Kot že vemo, naši podatki so vsebovali le statistični podatki o tekmah, medtem, ko je sam strel odvisen od igralca. Na primer, mi smo dobili podatek, da je določena ekipa imela določeno število strelov proti golu na določeni tekmi, nismo pa vedeli kdo in kolikokrat je streljal, kar je zelo pomemben podatek. Zaradi tega smo dobili najboljši rezultati, ko se je model učil za eno- letno obdobje. Razlog je preprost, saj igralci pogosto prestopajo v druge ekipe in npr nek igralec, ki je igral za določeno ekipo, je naslednje leto prestopil v drugo ekipo v drugo državo. Torej ta igralec, se ne nahaja v ekipi naslednje leto, vendar naš model tega ne ve in predvideva, da je ekipa ista kot lansko leto oziroma več let. Zato so rezultati za eno- letno obdobje najboljši, saj se ekipa manj spremeni v enem letu kot v dveh, treh ali pa celo štirih letih.



(a) RAE

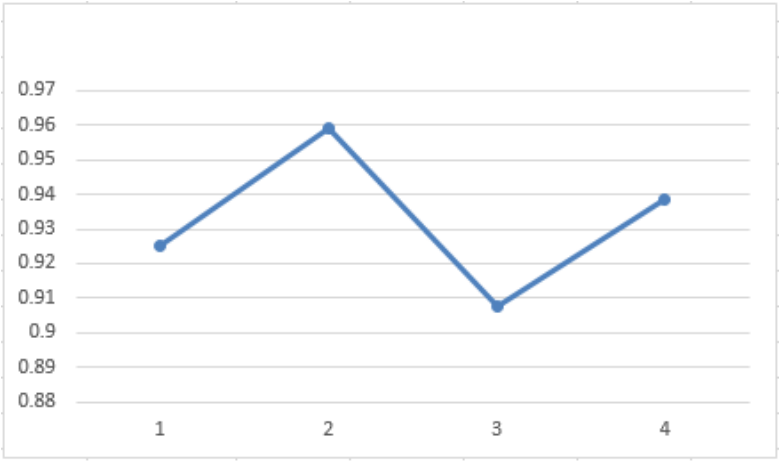


(b) MAE

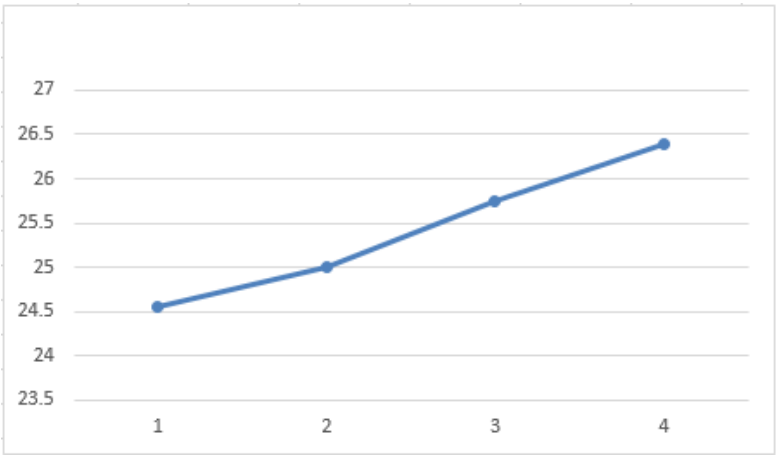
Slika 4.6: Primerjava napak za različna obdobja pri napovedovanju število strelov proti golu

Pri napovedovanju število kotov (glej slika 4.7) opazimo, da dobimo slabe rezultate. Očitno je, da se grafa razlikujeta, saj je RAE najnižja (model se zmoti v slabih 91%), ko se model uči na podatkih iz tri- letnega obdobja, dokler je MAE najnižja (model natačno v povprečju zgreši za 24,5 kotov), ko je učna množica podatkov za eno- letno obdobje. Razlog za slabši rezultat je podoben kot pri napovedovanju število strelov proti golu, kajti ta dva atributa sta povezana oziroma odvisna od igralca. Iz pravila nogometne igre vemo, da ekipa dobi kot takrat, ko igralec strelja proti golu nasprotne ekipe,

žoga pa se odbije od igralca iz nasprotne ekipe in gre iz igrišča v območje vodoravno s postavljenostjo gola. Kot vemo podatkov o igralcih nimamo in zato sta napaki zelo visoki, vendar obstaja izboljšanje v primerjavi z navadnim Mean algoritmom, ki je vedno neuporaben oziroma je njegova vrednost nad 1.



(a) RAE

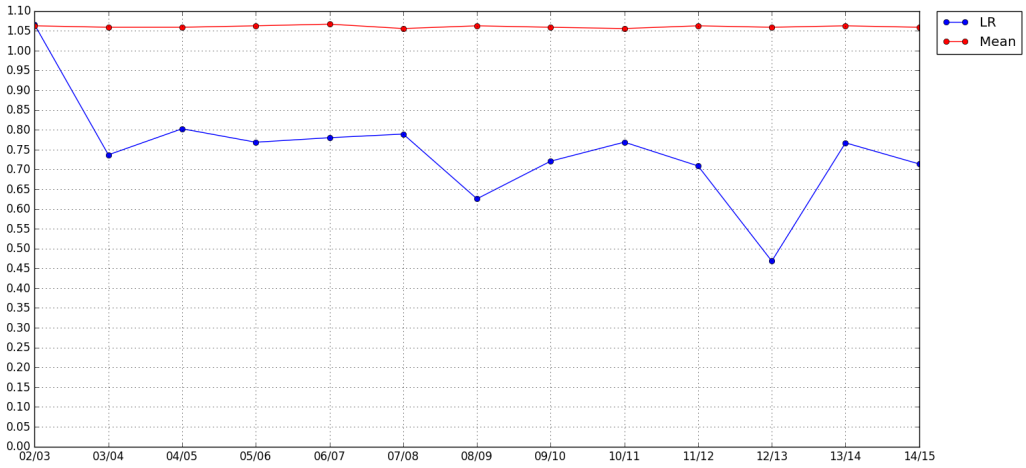


(b) MAE

Slika 4.7: Primerjava napak za različna obdobja pri napovedovanju število kotov

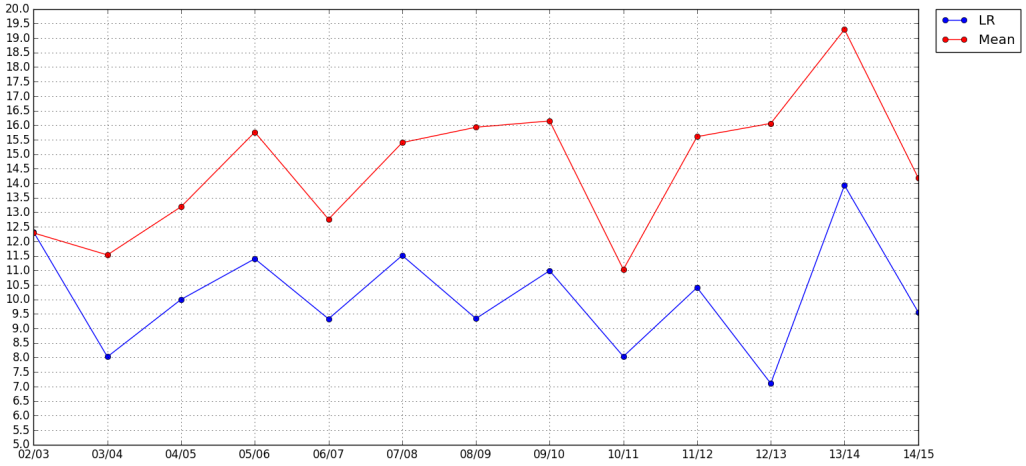
Od dosedanjih rezultatov lahko sklepamo, da smo najbolje napovedali število točk in le takrat, ko se je model učil na omrežju iz dvo- letnega

obdobja. Zato smo se odločili, da bomo tovrstno napovedovanje podrobneje opisali.



Slika 4.8: Napaka RAE za vsako napovedano leto posebej

Na Sliki 4.8 je predstavljeno, kako se napaka RAE obnaša skozi čas. Na osi x se nahaja napovedano leto, na osi y pa uspešnost napake. Iz grafa je razvidno za koliko algoritem LR zmanjša napako RAE v primerjavi z navadnim algoritmom Mean. V večini primerov se napaka giblje med 0.7 in 0.8, razen za leta 08/09, 12/13 in 02/03. To pomeni, da naš model v povprečju zgreši v 70 do 80% oziroma pravilno napove v 30 do 20%. Po drugi strani, sta prvi dve omenjeni izstopajoči leti za napovedni model izdatno učinkovitejši, posebno v letu 12/13, ko se RAE zniža do 47%. Razlog za boljši rezultat gre iskati v tem, da so nekatere vrednosti lastnosti BC, HUB in AUTH zelo visoke oziroma nizke, kar pomeni, da imamo nekaj ekip, ki so v tistih treh letih igrale konstantno dobro ali slabo. V našem primeru to pomeni, da ni nekih drastičnih sprememb na vrhu lestvice v primeru leta 08/09 in 12/13 oziroma dnu lestvice v letu 02/03.



Slika 4.9: Napaka MAE za vsako napovedano leto posebej

Pri merjenju napake MAE je zgodba podobna, saj algoritem LR izdatno zniža napako v primerjavi z navadnim algoritmom Mean. Rezultati niso za vsako leto sorazmerno enaki kot pri RAE, saj MAE ni odvisna od povprečnega števila točk vseh ekip (glej 2.2.2). Po drugi strani pa so rezultati pričakovani, saj se dejanska povprečna zgrešitev točk giblje med 7 in 11,5 točk, kar pomeni, da naš model v povprečju teoretično zgreši 3 do 4 tekme na ekipo.

Poglavje 5

Sklepne ugotovitve

Namen naloge je bil predstaviti nov, drugačen pristop za napovedovanje lastnosti nogometnih tekem. Ta se osredotoča na izgradnjo kakovostega omrežja in s pomočjo lastnosti iz analize teorija omrežij zgraditi uspešen napovedni model. Rezultati so nam pokazali, da je samo testiranje uspešnosti napovednega modela zapleten proces, oziroma časovno zahteven. Odločili smo se, da bomo model nenehno logično poenostavljali. Ugotovili smo, da je množica vhodnih podatkov zelo pomembna, kajti naš model je bil najučinkovitejši, ko je učna množica temeljila na podatkih iz eno ali dve letnega obdobja. V takih omrežij, nam je pri napovedovanju uspelo zmanjšati napako RAE do 70%, kar je bil najboljši rezultat, saj je pri napovedovanju števila strelov proti golu in kotov bila napaka večja za 10 oziroma 20%.
Rešitev za nadgradnjo napovednega modela se skriva v množici vhodnih podatkov. Ko bodo na voljo podatki o igralcih za posamezno tekmo, bi bila množica vhodnih podatkov za napovedni model večja in tudi bogatejša. Prav tako s podatki o igralcih, bi prvotno omrežje dajalo boljše lastnosti. To pomeni, da bi se izdatno znižale napake, kajti kot smo že povedali v rezultatih naš model ne ve kdo je zadel gol, kdo je strejlal proti golu, kdo je naredil rumeni ali rdeči karton ali kdo je naredil prekršek. Prav tako, ne vemo kdo je na določeni tekmi igral, saj so vsi naši ciljni napovedovalni atributi odvisni od igralcev. Nimamo podatka kdo je bil trener, kajti tudi on ima veliko in

pomembno vlogo v ekipi. Če bi imeli podatke o igralcih, bi vse to vedeli in upoštevali v naš napovedni model. Vendar tudi v primeru, če bi imeli vse te podatke, obstajajo dejavnosti, ki jih naš model ne bi vedel, na primer kakšno je počutje igralca, kakšne so njegove poškodbe, koliko je bilo privatnih odsotnosti itd. V nogometnem svetu tovrstnemu problemu strokovnjaki rečejo tekmovalnost. V primeru, da bi neka ekipa vedela skoraj vse o drugi ekipi, bi poznala tudi vse nasprotnikove slabosti in bi na lažji način zmagala. Takšne podatke je nemogoče pridobiti.

V tej nalogi se je izkazalo, da je bilo pomanjkanje podatkov verjetno ena od glavnih težav. Kljub temu smo uspešno znižali napake v primerjavi z navadnim pristopom, kar pomeni, da smo v veliki meri dosegli naše cilje.

Tabele

Slike

Literatura

[1] T. J. Archdeacon. *Correlation and Regression Analysis*, University of Wisconsin Press, 1994.

[2] M. Barthelemy, "Betweenness centrality in large complex networks", The European Physical Journal B, št. 38, zv. 2, str. 163-168, 2004.

[3] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D. U. Hwang "Complex networks: Structure and dynamics", Physics Reports, št. 424, zv. 4-5, str. 175-308, 2006.

[4] P. Bonacich, "Some unique properties of eigenvector centrality", Social Networks, št. 29, zv. 4, str. 555-564, 2007.

[5] CFinder. Dosegljivo:
http://www.cfinder.org

[6] P. Y. Chen, P. M. Popovich. *Correlation: Parametric and Nonparametric Measures*, SAGE Publications, 2002.

[7] P. J. Curley, "English Soccer's Mysterious Worldwide Popularity", Contexts, št. 15, zv. 1, str. 78-81, 2016.

[8] I. Derenyi, G. Palla, T. Vicsek, "Clique Percolation in Random Networks", Phys. Rev. Lett., št. 94, zv. 16, str. 16-29, 2005.

[9] C. H. Q. Ding, H. Zha, X. He, P. Husbands, S. D. Horst, "Link Analysis: Hubs and Authorities on the World Wide Web", SIAM Review, št. 46, zv. 2, str. 256-268, 2004.

[10] G. Dong, J. Pei. *Sequence Data Mining*, Springer Science & Business Media, 2007.

[11] N. Duhan, A. K. Sharma, K. K. Bhatia. "Page Ranking Algorithms: A Survey", v zborniku: Advance Computing Conference 2009. IACC 2009. IEEE International, 2009, str. 1530-1537.

[12] S. Fortunato, "Community detection in graphs", Physics Reports, št. 486, str. 75-154, 2010.

[13] L. C. Freeman, "Centrality in social networks: Conceptual clarification", Social Networks, št. 1, zv. 3, str. 215-239, 1978.

[14] Gephi. Dosegljivo:
https://gephi.org

[15] J. Han. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2001.

[16] R. A. Hanneman, M. Riddle. *Introduction to Social Network Methods*, University of California, 2005.

[17] Hubs and Authorities. [Online]. Dosegljivo:
http://documents.mx/documents/data-mining-and-semantic-web.html

[18] R. J. Hyndman, G. Athanasopoulos . *Forecasting: principles and practice*, Texts, 2013.

[19] M. Kuhn, K. Johnson. *Applied Predictive Modeling*, Springer, 2013.

[20] A. Lancichinetti, S. Fortunato, J. Kertesz, "Detecting the overlapping and hierarchical community structure in complex networks", New Journal of Physics, št. 11, zv. 3, 2009.

[21] A. N. Langville, C. D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press, 2011.

[22] A. Liaw, M. Wiener, "Classification and regression by randomForest", R news, št. 2, zv. 3, str. 18-22, 2002.

[23] Q. Liu, A. Puthenputhussery, C. Liu. "Novel general KNN classifier and general nearest mean classifier for visual classification", v zborniku: Image Processing (ICIP), 2015 IEEE International Conference on, 2015, str. 1530-1537.

[24] D. C. Montgomery, E. A. Peck, G. G. Vining. *Introduction to Linear Regression Analysis*, John Wiley & Sons, 2015.

[25] J. F. Navarro, C. S. Frenk, S. D. M. White, "A Universal Density Profile from Hierarchical Clustering", The Astrophysical Journal, št. 490, zv. 2, str. 490-493, 1997.

[26] M. E. J. Newman. *Networks: An Introduction*, Oxford university press, 2010.

[27] T. Opsahl, F. Agneessens, J. Skvoretz "Node centrality in weighted networks: Generalizing degree and shortest paths", Social Networks, št. 32, zv. 3, str. 245-251, 2010.

[28] Orange. Dosegljivo:
http://orange.biolab.si

[29] PageRank Algorithm. [Online]. Dosegljivo:
https://apavelescu.wordpress.com

[30] Pearson product-moment correlation coefficient. [Online]. Dosegljivo:
https://en.wikipedia.org/wiki/`Pearson_product-moment_correlation_coefficient`

[31] M. A. Porter, J. P. Onnela, P. J. Mucha, "Communities in Networks", SIAM Review, št. 56, zv. 9, str. 1082-1097, 1164-1166, 2009.

[32] A. M. Prasad, L. R. Iverson, A. Liaw, "Newer Classification and Regression Tree Techniques: Bagging and Random Forests for Ecological Prediction", Ecosystems, št. 9, zv. 2, str. 181-199, 2006.

[33] Programming language Python. Dosegljivo: https://www.python.org

[34] D. S. Putler, R. E. Krider. *Customer and Business Analytics*, CRC Press, 2015.

[35] Spearman's rank correlation coefficient. [Online]. Dosegljivo: https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient

[36] Stanford Network Analysis Project. Dosegljivo: http://snap.stanford.edu

[37] J. A. K. Suykens, J. Vandewalle, "Least Squares Support Vector Machine Classifiers", R news, št. 9, zv. 3, str. 293-300, 1999.

[38] R. J. Trudeau. *Introduction to Graph Theory*, Courier Corporation, 2013.

[39] J. Xie, S. Kelley, B. K. Szymanski, "Overlapping Community Detection in Networks: the State of the Art and Comparative Study", CoRR, št. 45, zv. 4, 2013.